Indian Institute of Technology Patna
CS1101- Foundations of Programming

Lab3: Data types and Expressions

Date: 18-8-2025

Task1: **Using the provided `base_convert` program, complete the table with your own sample numbers. One example entry has already been filled in for you. (take minimum of 15 sample values)**

Number	int	Binary (int)	Float (binary format)	Parsed Value (float)	Parsed Value (double)
-25.1	-25	(11111111111111111111111111100111)$_2$ (FFFFFFE7)$_{16}$	(01000001110010001100110011001101)$_2$ (41C8CCCD)$_{16}$	-25.1000004	-25.100000000000001

Task2: **Develop a C program to determine, for each standard data type, its size (in bytes), minimum and maximum values, and the appropriate `printf` format specifiers; use these results to complete the table**

Type	size	Max Value	Min Value	Format Spec.
char	1 byte	127	-128	%c/%d

……	……	……	……	……

//Sample Code

```
#include <stdio.h>
#include <limits.h>   // CHAR_MAX, SHRT_MAX, INT_MAX, LONG_MAX, LLONG_MAX, etc.
#include <float.h>    // FLT_MAX, DBL_MAX, LDBL_MAX, FLT_DIG, DBL_DIG, LDBL_DIG
#include <stdint.h>   // optional: fixed-width types (not used in the table)
int main(void) {
    /* Integer-family */
    printf("%-12s %-7zu %-28d %-18s\n", "char",        sizeof(char),        CHAR_MAX,      "%c / %d");
    printf("%-12s %-7zu %-28d %-18s\n", "short",       sizeof(short),       SHRT_MAX,      "%hd / %hu");
    printf("%-12s %-7zu %-28d %-18s\n", "int",         sizeof(int),         INT_MAX,       "%d / %u");
    printf("%-12s %-7zu %-28ld %-18s\n", "long",        sizeof(long),        LONG_MAX,      "%ld / %lu");
    printf("%-12s %-7zu %-28lld %-18s\n", "long long",   sizeof(long long),   LLONG_MAX,     "%lld / %llu");
    /* Floating-point family: For floats, the 'minimum' (DBL_MIN etc.) is the smallest positive normal number,
       not the most negative. Since we were asked for maxima, we print *MAX only*.  */
    printf("%-12s %-7zu %-28e %-18s\n", "float",       sizeof(float),       FLT_MAX,       "%f / %e / %g");
    printf("%-12s %-7zu %-28e %-18s\n", "double",      sizeof(double),      DBL_MAX,       "%lf / %e / %g");
    printf("%-12s %-7zu %-28Le %-18s\n", "long double", sizeof(long double), LDBL_MAX,      "%Lf / %Le / %Lg");
```

Task3: Given the following data declaration statements:

int A, B=2, C=3;

char D='A';

float E=257.8;

long Fran=0xFFFFE;

unsigned Gill=0177777;

what is the value in each variable after each of the following statements(Write a C program to evaluate)

(1) A = B * 2 + C / C;

(2) A = 10 / B + C;

(3) A = 12 / B * C;

(4) A = C / 4;

(5) A = C % 4;

(6) A = C % 4 - C;

(7) A = B << 2;

(8) A = B >> 2;

(9) A = B << 2 + C;

(10) A = 1 << 15 >> 15;

(11) D = 1 << 7 >> 7;

(12) A = C & 1;

(13) A = C | 1;

(14) A = C ^ 1;

(15) A = ~B;

(16) A = ~B & C;

(17) A = ((-1 ^ B) & 7) | B;

(18) A = Fran;

(19) A = B * Fran;

(20) A = C & Fran;

(21) D = Fran;

(22) Gill = Fran;

(23) Gill = -1;

(24) Fran = Gill;

(25) A = E;

(26) D = E;

(27) Fran = ~B;

(28) D = ~B;

(29) E = 4 / C;

(30) E = 4.0 / C;

(31) E = 4 / (float)C;

(32) Fran = (int)Gill;

(33) A = B++;

(34) A = -~--C;

(35) B = (C = 3) - 1;

(36) B += C;

(37) B |= ~B;

(38) A ^= ~A;

(39) B = 2 * ++Gill + 2;

(40) B %= C <<= 2;

(41) B += A &= ~A;

(42) A = C = D = E = Fran = 3;

(43) A = B++ + ++C;

(44) A = (Gill = 1) << 15 >> 15;

Task 4: Develop a program that adds and multiplies two complex numbers. Let the complex numbers be x=a+ib and y=c+jd ;

In Record

Task 1, Task 2, and Task 4 are to be recorded